CSS.203.1: Computational Complexity 17 Feb, 2021

Problem Set 1

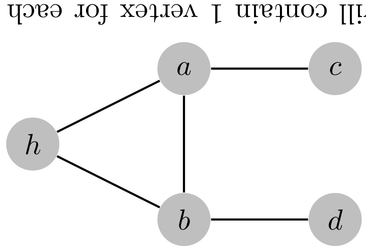
- Due Date: **3 Mar (Wed), 2021**
- Many of the problems are from the textbook "Computational Complexity: A Modern Approach" by Arora and Barak, and these are indicated on the side.
- Turn in your problem sets electronically (LaTeX, pdf or text file, or scanned pdf) on Acadly.
- Collaboration is encouraged, but all writeups must be done individually and must include names of all collaborators.
- Refering sources other than the text book and class notes is strongly discouraged. But if you do use an external source (eg., other text books, lecture notes, or any material available online), ACKNOWLEDGE all your sources (including collaborators) in your writeup. This will not affect your grades. However, not acknowledging will be treated as a serious case of academic dishonesty.
- The points for each problem are indicated on the side.
- Be clear in your writing.

1. **[P, NP, coNP, NP-complete, coNP-complete]** (10+3+6+6)

   Identify the smallest complexity class (among P, NP, coNP) in which each of the problem lies. Furthermore, if the problem is in NP (or coNP), mention if the problem is NP-complete or coNP-complete. In each case, substantiate your classification.

$$\begin{aligned}
\text{3-C}\textsc{olour} &= \{G \mid G \text{ can be coloured using 3 colours such that no edge is monochromatic } \}\\
\text{NON-ISO} &= \{(G,H) \mid \text{ there is no edge preserving bijection between } G \text{ and } H\}\\
\text{2SAT} &= \{\varphi \mid \varphi \text{ has at most 2 literals in any clause and is satisfiable. } \}\\
\text{BIPARTITE} &= \{G \mid G \text{ is a bipartite graph}\}
\end{aligned}$$

   [Hint: for 3-Colour: Reduce from 3-SAT. Your graph will contain 1 vertex for each literal, and 3 special vertices connected in a triangle (which must then be coloured with the three distinct colours, say T, F and Green). You may find the following gadget (see graph below) useful. Suppose that the head h and the two legs c and d are connected to the special vertex coloured Green, then any valid 3-colouring of this graph has the property that at least one of the two legs c or d has the same colour as the head h.]



2. (Problem 2.15 and 2.34) **[VERTEX COVER]** (8+5)

   (a) In the "VERTEX COVER" problem, you are given an undirected graph G as input and an integer k. The task is to decide whether there is a subset S of at most k vertices such that, for every edge $(i, j) \in G$, at least one of i or j is in S (such a set is called a *vertex cover*.

   Show that the VERTEX COVER problem is NP-complete.

   (b) Suppose you are given an undirected graph G and an integer k and are told that either (i) the smallest vertex cover of G is of size at most k, or (ii) it is of size at least $3k$.

   Show that there is a deterministic polynomial time algorithm to distinguish these two cases. Can you do it with a smaller constant than 3?

   Also, since you have just shown that VERTEX COVER is NP-hard, why does this algorithm not show that $\mathsf{P} = \mathsf{NP}$?

   [Hint: You may use the fact that there exists a deterministic polynomial time algorithm for matching.]

3. (Problem 2.16) **[MAXCUT]** (7)

   You are given an undirected graph G and integer k and you have to decide whether there is a subset S of vertices S such that there is at least k edges with one endpoint in S and one endpoint in $\overline{S}$ (the complement of S). Prove that this problem is NP-complete.

4. **[coNP]** (4+4+7)

   (a) (Problem 2.24) Here are two "different" definitions for the class coNP.

   Definition 1. *A language $L \subseteq \{0,1\}^*$ is in coNP if its complement $\overline{L}$ is in NP.*

   Definition 2. *A language $L \subseteq \{0,1\}^*$ is in coNP if there exists a polynomial $p : \mathbb{N} \to \mathbb{N}$ and a polynomial-time deterministic TM M such that for every $x \in \{0,1\}^*$ we have*

   $$x \in L \Leftrightarrow \forall u \in \{0,1\}^{p(|x|)} \ , \ M(x, u) = 1.$$

   Show that these two definitions are equivalent.

   (b) (Problem 2.25) Prove that if $\mathsf{P} = \mathsf{NP}$ then $\mathsf{NP} = \mathsf{coNP}$.

   (c) (Problem 2.33) Let $\Sigma_2\,\mathrm{SAT}$ denote the following decision problem: Given a quantified formula ψ of the form

   $$\psi := \exists_{x \in \{0,1\}^n} \forall_{y \in \{0,1\}^m} \varphi(x, y),$$

   where φ is a CNF formula, decide if ψ is true. That is, decide whether there is an x such that for all y we have $\varphi(x, y)$ is true.

   Prove that if $\mathsf{P} = \mathsf{NP}$, then $\Sigma_2\,\mathrm{SAT} \in \mathsf{P}$.

5. **[Unary languages and NP]** (8+7)

   (a) (Problem 2.30) A language is said to be *unary* if every string in the language is of the form 1^i (the string of i ones) for some $i > 0$. Show that if some unary language L is NP-complete, then $\mathsf{P} = \mathsf{NP}$.

   (b) (Problem 2.32) Prove that if *every* unary language in NP is also in P, then $\mathsf{EXP} = \mathsf{NEXP}$.

   [Hint: Problem 2.30: Besides the hint given in the textbook, use the fact that for any m, the number of strings in a unary language L of length at most m is at most m.]

6. (Problem 2.5) **[PRIMES $\in$ NP]** (10)

   Let PRIMES $= \{\ulcorner n \urcorner \mid n \in \mathbb{N} \text{ is prime}\}$ where $\ulcorner n \urcorner$ is just the binary representation of the integer n. Show that PRIMES $\in \mathsf{NP}$.

   (For the purposes of this question, you are not allowed to merely say PRIMES $\in \mathsf{P}$ without a proof. :-))

   [Hint: You can use the following fact: n is prime if and only if for every factor q of $n-1$, there is a number $a \in \{1, 2, \ldots, n-1\}$ that satisfies $a^{n-1} = 1 \bmod n$ but $a^{(n-1)/q} \neq 1 \bmod n$.]

7. [Circuit-SAT **is NP-complete**] (8+7)

   A circuit C on n inputs is a directed acyclic graph with n *sources* (vertices with no incoming edges) and one *sink* (vertex with no outgoing edges). All nonsource vertices are called *gates* and are labeled with one of $\vee, \wedge$ or $\neg$. The vertices labeled with $\vee$ or $\wedge$ have fan-in 2 and the vertices labeled with $\neg$ have fan-in 1. If C is a Boolean circuit and $x \in \{0,1\}^n$ is some input, then the output of C on x, denoted by $C(x)$, is defined in the natural way. A circuit C is said to be satisfiable if there exists a x such that $C(x) = 1$.

   $$\textsf{Circuit-SAT} = \{C \mid C \text{ is satisfiable}\}$$

   (a) Prove (along the lines of the proof of the Cook-Levin Theorem discussed in lecture) that Circuit-SATis NP-hard. (don't reduce SAT to Circuit-SAT).

   [Hint: Consider a binary encoding of every snapshot z_i. Show that there exists a circuit that computes snapshot z_i given snapshots $z_i, z_{\mathrm{prev}(i)}$ and y. Compose these circuits suitable to get the final result.]

   (b) Prove that Circuit-SAT$\leq_p$ 3SAT (this gives an alternate proof of the Cook-Levin Theorem).